Reversible universal quantum computation within translation invariant systems

K. G. H. Vollbrecht[1] and J. I. Cirac[1]
1 *Max-Planck Institut für Quantenoptik, Hans-Kopfermann-Str. 1, Garching, D-85748, Germany*
(Dated: February 1, 2008)

We show how to perform reversible universal quantum computation on a translationally invariant pure state, using only global operations based on next-neighbor interactions. We do not need not to break the translational symmetry of the state at any time during the computation. Since the proposed scheme fulfills the locality condition of a quantum cellular automata, we present a reversible quantum cellular automaton capable of universal quantum computation.

PACS numbers: 03.75.Lm, 03.67.Lx, 42.50.-p

I. INTRODUCTION

The great challenge in quantum information theory and quantum computation is to build a quantum computer, which is conjectured to be exponentially faster than its classical counterpart. Many proposals have been presented in recent years that claim a physical system to be a good candidate for doing quantum computation. In order to be able to build a quantum computer, the proposed scheme has to satisfy a checklist [1] of required properties that are widely believed to be necessary for universal quantum computation. Two of the points on this checklist are, that a qubit has to be encoded into a well defined physical system, i.e., there should be two levels representing $|0\rangle$ and $|1\rangle$, and that each of these systems should be manipulated individually. We present a quantum computation scheme that seems to get around these requirements, using translationally invariant states and global addressing.

To encode several qubits into a translationally invariant system seems to be contradictory, because by definition all individual systems are identical and can not carry different kinds of quantum information. Restricting to global operations that are themselves translationally invariant, we are not able to break the translation symmetry, in particular, we can not address a single system alone.

But such a situation occurs for example in optical lattices, for which it is experimentally a very hard task to address single sites in the lattice. In the ideal situation, this lattice is prepared in a translationally invariant state, with exactly one atom per site [2], all in the same internal state. Schemes for quantum computation that have been proposed for this system require breaking the translational symmetry as a first step [4, 5, 6] , e.g., by using imperfections in the lattice [3].

In this paper we introduce a novel method of performing quantum computations on a translationally invariant, one dimensional lattice of 5-level systems, that can be associated for example with a line of atoms in an optical lattice. Our scheme will be completely based on global operations, that can be carried out by reflectionally symmetric and translationally invariant next-neighbor Hamiltonians. Beside its physical relevance these kind of operations allow us to connect quantum computation to the context of quantum cellular automata [7, 8, 9].

Our system will stay in a pure, reflectionally symmetric and translationally invariant state during the whole computation. The main idea is based on the notion of ensemble quantum computation and is related to the schemes presented in [3].

The paper is organized as follows: in the first step we will introduce a quantum computation scheme that requires non-translationally invariant states. In the next step, we will show how we can get rid of this condition by switching to an ensemble quantum computation scheme. In a last step we will verify, that we can do this with a pure state.

II. PRELIMINARIES AND NOTATION

Assume we have atoms with 6 internal states $(|0\rangle \dots |5\rangle)$ in a one dimensional lattice of length m, where we assume m to be very large. To avoid any discussion about border effects, we will further assume either periodic boundary conditions or $m \to \infty$.

H_{xa}^{yb} denotes a reflection symmetric next-neighbor Hamiltonian, which is constructed in the following way: $h_{xa}^{yb} = |x,a\rangle\langle y,b| + h.c.$ is a Hamiltonian having the ability to transform $|xa\rangle$ into $|yb\rangle$ and vice versa, in the sense that $e^{ih_{xa}^{yb}t}|xa\rangle = i|yb\rangle$ for $t = \pi/2$. Here x, y denote two states in one site of the lattice and a, b two states in a neighboring site. In most of the cases x, y, a, b will be chosen to be one of the basis states $(|0\rangle \dots |5\rangle)$.

To make this Hamilton reflection symmetric we take $h_{(xa,yb)} = h_{xa}^{yb} + h_{ax}^{by}$. To our system we will apply translation invariant next neighbor Hamiltonians of the form

$$H_{xa}^{yb} = \sum_i h_{(xa,yb)}^{(i,i+1)}, \tag{1}$$

where $h_{(xa,yb)}^{(i,i+1)}$ denote the Hamiltonian $h_{(xa,yb)}$ applied on the ith and $(i+1)$th place. In particular, we will only apply Hamiltonians of the form H_{xa}^{xb}, with $\langle x|a\rangle = \langle x|b\rangle = 0$. In this case all $h_{(xa,xb)}^{(i,i+1)}$ commute with each other. Note, that the state a, b do not have to be orthogonal.

By U_{xa}^{xb} we denote the unitary operation $e^{iH_{xa}^{xb}t}$ with $t = \pi/2$ is chosen in such a way that every "isolated"

$|xa\rangle$ in the lattice is transformed into $|xb\rangle$ and vice versa, in the sense that $U_{xa}^{xb}|\ldots xac\ldots\rangle = i|\ldots xbc\ldots\rangle$ $(c \neq x)$. Without loss of generality we will ignore the factor i in the following. Since all terms in (1) commute (for $x = y, x \neq a, b$) we can calculate the action of U_{xa}^{xb} onto the lattice by applying twice the unitary operation

$$U_{xab} \otimes U_{xab} \ldots \otimes U_{xab} \tag{2}$$

but the second time shifted by one lattice site, where U_{xab} is a two qubit unitary operation defined as

$$U_{xab}|xa\rangle = |xb\rangle, U_{xab}|xb\rangle = |xa\rangle,$$

$$U_{xab}|ax\rangle = |bx\rangle, U_{xab}|bx\rangle = |ax\rangle.$$

Note, that this implies a finite propagation speed of quantum information, of one lattice site per operation, within the lattice, i.e., any manipulation on one site can effect another site after x operations at the earliest, if the two sites are x sites apart. Note that this is exactly the kind of condition that defines a system to be a quantum cellular automaton [7]. By V_{xy} we denote a unitary operation, that exchanges the levels x and y in every site of the lattice, i.e., $V_{xy} = \tilde{V}_{xy}^{\otimes m}$, with $\tilde{V}_{xy}|x\rangle = |y\rangle, \tilde{V}_{xy}|y\rangle = |x\rangle$.

III. A QUANTUM COMPUTER SCHEME

Now we show how to carry out a quantum computation using only translation and reflection invariant next-neighbor interactions, e.g., using only operations of the form U_{xa}^{xb}, V_{xy}, but assuming, in a first step, that the lattice is prepared in the non-translation invariant state

$$|230000000000000000\ldots\rangle. \tag{3}$$

The $|23\rangle$ at the beginning is called the pointer, every following "zero" will considered to be a qubit with the possible values $|0\rangle, |1\rangle$. The trick of doing quantum computation in this picture is that we will use Hamiltonians that somehow are located due to the pointers, e.g. the unitary U_{30}^{31} have only an effect to the qubit next to the "3" pointer. For the above state only the first qubit is affected, the state transforms into

$$|231000000000000000\ldots\rangle. \tag{4}$$

To address other qubits than the first one, we have to "move" the pointer around. This can be done in the following way:

- moving the pointer one site to the right (one extra level 4 required): By the sequence $S_{\rightarrow}$: U_{03}^{04}, U_{40}^{42}, U_{04}^{03}, U_{13}^{14}, U_{41}^{42}, U_{14}^{13}, V_{23}, the pointer $|23\rangle$ is shifted one position to the right. The qubit right of the pointer is shifted at the same time two sites to the left.

  To shift the pointer one position to the left we just apply the protocol backwards.

By moving the pointer to a special qubit and using Hamiltonians that are "located" by the pointer, we can apply arbitrary unitary operations to any qubit.

- local operations: First we move the pointer to the position left of the qubit we want to adress. Then we apply Hamiltonians of the form H_{3x}^{3y}, where x,y can be any qubit state $\alpha|0\rangle + \beta|1\rangle$. Doing this we can apply any one-qubit-unitary on the qubit .

To do quantum computation we need controlled operations, like a C-NOT gate. For this task, it is sufficient to have a C-NOT gate acting only between neighboring sites.

- controlled operations (CNOT) between two neighboring sites: By the sequence U_{02}^{04},U_{43}^{42},U_{21}^{20},U_{43}^{42},U_{21}^{20},U_{02}^{04} we apply a CNOT gate between the two qubits on the left and on the right site of the pointer. The qubit lying near the "2" is the source, the qubit near the "3" the target. To exchange the role of source and target we just have to exchange the role of "2,3" in the sequence. Since the pointer can be moved, we can apply this operation to arbitrary neighboring two-qubits.

Local operations and CNOT between two neighboring sites form a universal set of quantum gates, which enable us to do any kind of quantum computation. Finally, we need to measure a single qubit.

- measurement: First we move the pointer to the qubit we want to measure. Then we apply U_{31}^{34} and count the sites found in $|4\rangle$, i.e., we will find either zero ore one atom in state $|4\rangle$.

A. ensemble quantum computation

Using this kind of quantum computation scheme we can do ensemble quantum computations, i.e., we can run several quantum computers in parallel. If we start with several pointers, e.g. the state

$$|230000000002300000000230000000\ldots\rangle$$

we will have three quantum computers running in parallel. We only have to take care, that the distance between two pointers is larger than the number of qubits n used in (one copy) of the quantum computer.

In an ensemble quantum computation scheme a measurement takes place in all quantum computers at the same time. The number of atoms found in state $|4\rangle$ is given by the expectation value $< M_4 >= tr\rho M_4$ and can be compared to the number of atoms found in state $|3\rangle$, $< M_3 >= tr\rho M_3$, i.e., to the number of quantum computers, with

$$M_x = \sum_i P_x^{(i)}, \tag{5}$$

where $P_x^i = |x\rangle\langle x|$ applied on the ith site. Note that this measurement is again a global operation.

We can use pointers of the form $|23\rangle$ and $|32\rangle$ at the same time, e.g.

$$|230000000000000000000320000000023000000\ldots\rangle.$$

The pointers of the form $|23\rangle$ will address the qubits to their right, the $|32\rangle$ the qubits to their left.

B. Starting from a mixed translational invariant states

We will now show, that we can perform ensemble quantum computation starting from a translation invariant state. For simplicity, we first start with a mixed state, where every site is either in $|0\rangle$ or in $|5\rangle$, with corresponding probabilities $1-\varepsilon$, ε. So we start with a mixed state

$$\rho = \sum_i p_i |\phi_i\rangle\langle\phi_i|, \tag{6}$$

where $|\phi_i\rangle$ is the set of all possible states only consisting of $|0\rangle$ and $|5\rangle$, e.g., the state

$$|000000500000000000005000500000000 5\rangle. \tag{7}$$

The sum goes over all possible configurations of $|5\rangle$ and $|0\rangle$, with the corresponding probabilities $p_i = \varepsilon^{\#|5\rangle}(1-\varepsilon)^{\#|0\rangle}$. This mixed states is a translation invariant and can be for example produced from the state $|0\rangle^{\otimes n}$ by adding some incoherent translation invariant noise that is mixing $|0\rangle$ and $|5\rangle$.

To do quantum computation using above scheme we need to create pointers.

C. Creating pointers

By the sequence U_{50}^{52},U_{20}^{23},U_{32}^{34}, U_{50}^{52} and U_{32}^{34}, we create $|23\rangle$ and $|32\rangle$ pointers. The $|5\rangle$ states cuts the lattice in partitions of different length. Every such partition with more than 4 $|0\rangle$ gets a $|23\rangle$ and $|32\rangle$ at the borders, while partitions with less than 3 $|0\rangle$ stays unchanged. E.g., state (7) transforms into

$$|000032523000000000325000523000032 5\rangle. \tag{8}$$

Each large enough partition contains two quantum computers; each of them can address half of the qubits in the partition. Note, if you move the pointers too far, so that they meet each other, they just cross each other. If the length of the partition is even and we apply $S_{\rightarrow}$ on $|2332\rangle$ we get $|3223\rangle$. In the odd case, the meeting pointers first get into an "inactive" state, and in the next step pass each other. In particular, if we apply two times $S_{\rightarrow}$, we get $|23032\rangle \rightarrow |04040\rangle \rightarrow |32023\rangle$ and $|23132\rangle \rightarrow |14141\rangle \rightarrow |32123\rangle$. Note, that the states $|04040\rangle, |14141\rangle$ are "inactive", in the sense that they are not affected if we apply a CNOT or a local operations with the Hamilton H_{3x}^{3y}.

If we try to move the pointer over the $|5\rangle$ border the pointers change their direction, i.e., $|235\rangle \rightarrow |325\rangle$. During any kind of quantum computation, the pointers can not leave their partition and the total number of pointers stays constant.

D. Signal and scalability

Every $|00500\rangle$ splits up into two quantum computers "23" and "32", one running to the left and one running to the right. If we use this for ensemble quantum computation with a algorithm using n qubits, we get in the end of the protocol a correct signal from all partitions of length greater than $2n+4$. On smaller partitions the algorithm will not run correct, because the two pointers in the partition will address qubits belonging to the other pointer.

We can control the number of correctly working quantum computers and incorrectly working quantum computers by the probability ε.

The expected total number of partitions found in a lattice of size m generated in this way is given by $m\varepsilon$. Each of this partitions has a random length, where the probability having exactly n $|0\rangle$-sites is given by $(1-\varepsilon)^n\varepsilon$. The probability that a partitions has n or more sites is given by $(1-\varepsilon)^n$.

Therewith, the expected number of partitions of length bigger than $2n+4$ on a m site lattice is given by

$$\#working = m\varepsilon(1-\varepsilon)^{2n+4},$$

where $m\varepsilon$ is the total number of partitions found in the lattice and $(1-\varepsilon)^{2n+4}$ is the probability that such a partition is bigger than $2n+4$. The ratio of long enough partitions to total number of partitions is then just $(1-\varepsilon)^{2n+4}$. If we choose the probability ε small enough, then the ratio is going to one, i.e., the signal of the measurement will only come from correctly working quantum computers. But choosing ε to be very small can result in a scalability problem for our scheme, because ε is responsible for the total number of quantum computers found in the lattice. Fortunately, we have to decrease ε only polynomially with the number qubits n. Choosing $\varepsilon \sim 1/n^2$ we get

$$\#working = m\frac{1}{n^2}(1-\frac{1}{n^2})^{2n+4} \sim m\frac{1}{n^2}.$$

This means, that the probability of a quantum computer to be working goes to one, while the total number of quantum computers decrease only polynomially with $\frac{1}{n^2}$.

E. Starting from pure translational invariant states

Instead of using a mixed translation invariant state, we can start with the pure state

$$|0\rangle^{\otimes m}. \tag{9}$$

This can be transformed into

$$|\Phi\rangle = (\sqrt{1-\varepsilon}|0\rangle + \sqrt{\varepsilon}|5\rangle)^{\otimes m}. \tag{10}$$

by applying the same unitary on every site, which is obviously a translation invariant unitary operation. We claim, that the above procedure starting with a translational invariant mixed state works in exactly the same way for this translation invariant pure state.

Instead of having a mixture of states, we now have the coherent superposition of exactly the same states, i.e.,

$$|\Phi\rangle = \sum_i \sqrt{p_i}|\phi_i\rangle, \tag{11}$$

where p_i and $|\phi_i\rangle$ are as in (6). During a quantum computation the state transforms to

$$|\Phi'\rangle = \sum_i \sqrt{p_i}|\phi_i'\rangle. \tag{12}$$

For the final measurement we get

$$\langle\Phi'|M_x\Phi'\rangle = \sum_{ij}\sqrt{p_i p_j}\langle\phi_i'|M_x\phi_j'\rangle = \sum_i p_i\langle\phi_i'|M_x\phi_i'\rangle. \tag{13}$$

The second equality is due to the fact, that the operations and measurements do not affect the $|5\rangle$ levels. This guarantees that $\langle\phi_i'|M_x\phi_j'\rangle = \delta_{ij}\langle\phi_i'|M_x\phi_i'\rangle$. The measurement gives the same results, as if we would have used the state (6), for which we already proved our scheme to work.

F. Using only 5 internal levels.

We can reduce our scheme using only 5 internal states. For preparing the pointer we just replace the role of the $|5\rangle$ by a $|1\rangle$, i.e., we transform randomly distributed $|00100\rangle$ into $|32123\rangle$. Using this kind of scheme the pointers can break out of their partition and effect the computation in neighboring partitions. But this turns out to be no problem, because by choosing $\varepsilon \sim 1/n^2$ the probability for this goes to zero compared to the probability of correctly working quantum computers.

Using the pure state scheme we get the problem, that we can not ensure any more, that $\langle\phi_i'|M_x\phi_j'\rangle = \delta_{ij}\langle\phi_i'|M_x\phi_i'\rangle$, since the $|1\rangle$ configuration, that defines the states, can be changed due to the incorrect working quantum computers. Instead of using the $|1\rangle$ configuration, we take now the configuration of the pointers and use the same kind of argumentation. Let us take the state

$$|\Phi\rangle = (\sqrt{1-\varepsilon}|0\rangle + \sqrt{\varepsilon}|1\rangle)^{\otimes m}. \tag{14}$$

and apply the pointer creation protocol U_{10}^{12},U_{20}^{23},U_{32}^{34}, U_{10}^{12} and U_{32}^{34}. We can write the resulting states as

$$|\Phi\rangle = \sum_i \sqrt{q_i}|\psi_i\rangle, \tag{15}$$

where i labels all possible pointer configurations that can occur due to the protocol, i.e., all possible configurations of $|23\rangle, |32\rangle$. This is basically the same decomposition of the state $|\Phi\rangle$ as in (11), with the difference that all sets of states $\{|\phi_i\rangle\}$ that differ only at partitions that are too small to create any pointer are merged together to one $|\psi_k\rangle \sim \sum_i p_i|\phi_i\rangle$, e.g. the two states

$$\alpha|\ldots 00321112300\ldots\rangle, \beta|\ldots 00321012300\ldots\rangle$$

have different $|1\rangle$ configuration, i.e., they correspond ($|1\rangle$=$|6\rangle$) in (11) to to differen $|\phi_i\rangle$. In (15) they are represented now in only one term,

$$|\ldots 0032\rangle|\alpha 101 + \beta 111\rangle|2300\ldots\rangle,$$

because they have the same pointer configuration. All working quantum computers in such a $|\psi_i\rangle$ are correctly initialized with only zeros. Superpositions, like $|\alpha 101 + \beta 111\rangle$, appear only in areas that are not addressed by a working quantum computer. There are no pointers destroyed during the computation or the measurement. This guarantees now, that we get $\langle\psi_i'|M_x\psi_j'\rangle = \delta_{ij}\langle\psi_i'|M_x\psi_i'\rangle$ and therewith

$$\langle\Phi'|M_x\Phi'\rangle = \sum_i q_i\langle\psi_i'|M_x\psi_i'\rangle. \tag{16}$$

The measurement outcomes are the same as if we would have started with the mixed state $\rho = \sum_i q_i|\psi_i\rangle\langle\psi_i|$. The number of correctly working to incorrectly working quantum computers is the same as in (6), i.e., we can effectively suppress the signal coming from incorrectly working quantum computers.

IV. CONCLUSION

We have presented a scheme to perform universal quantum computation within a translationally and reflection invariant system. This scheme requires 5 internal levels at each position, global-local operations and next-neighbor interactions. The presented scheme is scalable in the sense that the number of resources, i.e. sites, scales quadratically [10] with the number of qubits used during the quantum computation. We formulate our scheme in the language of Hamiltonians, but we could have started from unitaries which are translationally and reflection invariant. Apart from the obvious relation to some physical implementation, like atoms in optical lattices, our result shows it is possible to perform quantum computation in translationally invariant systems. Furthermore, in the

case of unitaries, what we have built is a quantum cellular automaton. Thus our result implies that the power of a quantum cellular automata is equivalent to that of quantum computers. An open question is whether it is possible to achieve the same results with qubits, or if it is strictly necessary to use more than two levels per site. In the latter case, this would show that higher dimensional systems are indeed more powerful than qubits for certain quantum informational tasks.

We thank E. Solano for useful discussion. This work was supported by EU through RESQ and QUPRODIS projects and the Kompetenznetzwerk Quanteninformationsverarbeitung der Bayrischen Staatsregierung.

[1] D. P. DiVincenzo, Fortschritte der Physik **48**, 771 (2000).
[2] D. Jaksch, C. Bruder, J. I. Cirac, C. W. Gardiner, and P. Zoller, Phys. Rev. Lett. **81**, 3108 (1998).
[3] K. G. H. Vollbrecht, E. Solano, and J. I. Cirac, Phys. Rev. Lett. 93, 220502 (2004)
[4] T. Calarco, U. Dorner, P. Julienne, C. Williams, P. Zoller, quant-ph/0403197 (2004).
[5] A. Kay, J. K. Pachos, New J. Phys. 6 126 (2004)
[6] G. Ivanyos, S. Massar, A. B. Nagy, quant-ph/0502142(2005)
[7] B. Schumacher, R.F. Werner, quant-ph/0405174, (2004).
[8] S. Lloyd, quant-ph/9912086 (1999).
[9] S. C. Benjamin, Phys. Rev. A **61** 020301 (2000).
[10] Note, that so far we assume no errors occurring during the quantum computation.